SAND2011-4690C

CALCULATING PATH-DEPENDENT TRAVEL TIME PREDICTION VARIANCE AND COVARIANCE FOR A GLOBAL TOMOGRAPHIC P-VELOCITY MODEL

Jim Hipp[1], Andre Encarnacao[1], Chris Young[1], Sandy Ballard[1], Marcus Chang[1], Scott Phillips[2] and Mike Begnaud[2]

Sandia National Laboratories[1] and Los Alamos National Laboratory[2]

Sponsored by the National Nuclear Security Administration

Award Nos. DE-AC04-94AL85000/SL09-3D_Earth-NDD02[1] and DE-AC52-06NA25396/LA09-IRP-NDD02[2]

ABSTRACT

Several studies have shown that global 3D models of the compression wave speed in the Earth's mantle can provide superior first P travel time predictions at both regional and teleseismic distances. However, given the variable data quality and uneven data sampling associated with this type of model, it is essential that there be a means to calculate high-quality estimates of the path-dependent variance and covariance associated with the predicted travel times of ray paths through the model. In this paper, we show a methodology for accomplishing this by exploiting the full model covariance matrix.

Typical global 3D models have on the order of 1/2 million nodes, so the challenge in calculating the covariance matrix is formidable: 0.9 TB storage for 1/2 of a symmetric matrix, necessitating an Out-Of-Core (OOC) blocked matrix solution technique. With our approach the tomography matrix (G which includes Tikhonov regularization terms) is multiplied by its transpose (G^TG) and written in a blocked sub-matrix fashion. We employ a distributed parallel solution paradigm that solves for $(G^TG)^{-1}$ by assigning blocks to individual processing nodes for matrix decomposition update and scaling operations. We first find the Cholesky decomposition of G^TG which is subsequently inverted. Next, we employ OOC matrix multiplication methods to calculate the model covariance matrix from $(G^TG)^{-1}$ and an assumed data covariance matrix. Given the model covariance matrix we solve for the travel-time covariance associated with arbitrary ray-paths by integrating the model covariance along both ray paths. Setting the paths equal yields the variance for that path.

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

OBJECTIVES

The development of global 3D P-velocity models capable of producing high-quality travel time predictions is a significant accomplishment, but such models cannot be routinely used for locating seismic events until appropriate, path dependent uncertainty estimates are also available. In this paper, we show a methodology for calculating travel time variance and covariance for ray paths through 3D velocity models by exploiting the full model covariance matrix, a product of the tomograhic inversion process. For all but the crudest models, this is a computationally intensive calculation, necessitating the use of Out-Of-Core (OOC) solution techniques, which add an additional layer of complexity to the problem. We have developed OOC methods for calculating both the full model covariance matrix and the travel time variances and covariances. We demonstrate our methods using the most recent version of the SNL/LANL SALSA3D global 3D tomographic P-velocity model.

RESEARCH ACCOMPLISHED

Introduction

The ready availability of large data sets and powerful computers has made the production of 3D global tomographic geophysical models increasingly common. These models can provide important insight into the structure of the Earth, but more importantly for the monitoring community, they can also improve the quality of our seismic event locations by providing higher fidelity predictions of the observables used to locate events. Perhaps of most direct use for seismic monitoring are global 3D P-velocity models that can be used to predict first P travel times, the observables upon which seismic locations are generally based. Many such models have been produced over the past few decades, but only recently has it been clearly demonstrated that such models can provide superior travel time predictions over broad geographic regions (Ballard et al., 2010; Simmons et al., 2011). This is an important milestone, and the routine use of 3D models for event location now seems inevitable, but there are still important problems that must be addressed. Arguably the most important of these is the calculation of variance and covariance associated with travel time predictions through these models. For tomographic models especially, the uncertainty associated with different ray paths can be vary widely, reflecting the data coverage. Hence, travel times calculated through such a model are of limited utility without accompanying uncertainty estimates.

In the following sections we develop the theory for calculating travel time variance and covariance for paths through 3D Earth models, and then we apply our theory to the most recent version of the SNL/LANL SALSA3D model. We first review the calculation of the model covariance matrix for a tomographic velocity model, and then show how this matrix can be used to calculate variance and covariance for arbitrary ray paths through the model.

Calculation of the Model Covariance Matrix

Assume that we have a simple ray-theory tomographic problem of the form:

$$A\Delta s = \Delta d \tag{1}$$

where A is the $M \times N$ matrix of path geometries (the "data kernel"), Δs is the $N \times 1$ array of slowness changes, and Δd is the $M \times 1$ array of data residuals (observations - predictions). M is the number of observations, and N is the number of slowness nodes in our model. The typical problem is both over and under-determined, so we solve using damped least-squares. The damping implies adding additional constraints that attempt to impose zero change in the slowness solution:

$$\begin{bmatrix} A \\ \alpha I \end{bmatrix} \Delta s = \begin{bmatrix} \Delta d \\ 0 \end{bmatrix} \tag{2}$$

where α is the damping factor, I is an $N \times N$ identity matrix, and the 0 on right hand side is an $N \times 1$ array of zeros. Thus, the right hand side is now a $(M + N) \times 1$ extended data vector. The modified inversion is now trying to solve for slowness changes that better fit the data, and at the same time it is trying to minimize those changes. For paths with travel times not fit well by the starting model, these goals are in competition and the solution will be a compromise. A higher damping value amplifies the zero slowness change constraints, and hence leads to models that deviate less from the starting model.

Let $G = \begin{bmatrix} A \\ \alpha I \end{bmatrix}$ from which (2) can be re-written as

$$G\Delta s = \begin{bmatrix} \Delta d \\ 0 \end{bmatrix} \tag{3}$$

To invert, we first multiply both sides by the transpose of the bracketed matrix on the-left hand side:

$$G^T G\Delta s = G^T \begin{bmatrix} \Delta d \\ 0 \end{bmatrix} \tag{4}$$

At this point, a simplification is generally made to the right-hand side: the portion of the extended data array that is zero multiplies with the αI within G, cancelling it out, so (4) can be written as

$$G^T G\Delta s = A^T \Delta d \tag{5}$$

The solution follows as

$$\Delta s = (G^T G)^{-1} A^T \Delta d \tag{6}$$

Further, using (1) to substitute for Δd with Δs^{true} representing the true solution, we get

$$\Delta s = (G^T G)^{-1} A^T \, A\Delta s^{true} = R\Delta s^{true} \tag{7}$$

where R is the $N \times N$ model resolution matrix.

From the definition of covariance for a linear operator, one finds the expression for model covariance to be

$$\mathrm{C_M} = (\mathrm{G^T G})^{-1} \mathrm{G^T} \begin{bmatrix} \mathrm{C_{\Delta d}} & 0 \\ 0 & \mathrm{C_{\Delta s}} \end{bmatrix} \mathrm{G} (\mathrm{G^T G})^{-1} \tag{8}$$

where

$$C_{\Delta d} = \begin{bmatrix} \sigma^2_{\Delta d_0} & & & 0 \\ & \sigma^2_{\Delta d_0} & & \\ & & \ddots & \\ & & & \sigma^2_{\Delta d_{M-2}} \\ 0 & & & \sigma^2_{\Delta d_{M-1}} \end{bmatrix} \quad \text{and} \quad C_{\Delta s} = \begin{bmatrix} \sigma^2_{\Delta s_0} & & & 0 \\ & \sigma^2_{\Delta s_0} & & \\ & & \ddots & \\ & & & \sigma^2_{\Delta s_{N-2}} \\ 0 & & & \sigma^2_{\Delta s_{N-1}} \end{bmatrix} \tag{9}$$

are the data and delta slowness covariances, respectively. In both cases, we assume that individual variances are uncorrelated as indicated by the fact that all off-diagonal terms in Equation 9 are zero.

Note that in Equation 8 we do not assume that the variance of delta slowness in the model is zero, as is typically done with traditional approaches. This is justified given that the whole point of tomographic inversion is to solve for *non-zero* slowness adjustments, implying that there is some acceptable tolerance around the desired slowness adjustment value of zero. The variance of delta slowness should depend on the amount of damping and on the resolution of the particular node. High damping should lead to low variance, and well-resolved nodes should have a high-variance to allow the tomography to adjust the slowness as needed to fit the data.

We have set up some simple 2D straight ray tomographic problems and verified that model covariance estimates calculated using (8) (and hence assuming non-zero $\sigma^2_{\Delta s_i}$) provide more reasonable values than are otherwise attained with more traditional approaches where $\sigma^2_{\Delta s_i}$ are taken to be zero. Note that our proposed modification in the above derivation only affects the calculation of the model covariance matrix; the calculations of the tomographic model itself and of the model resolution matrix are the same.

Path-Dependent Travel Time Uncertainty Derivation

Once we have the model covariance matrix calculated, we can follow the standard methodology (e.g. Rodi & Myers, 2006), to calculate variance and covariance for ray paths through the model.

$$\sigma_{TT_{ij}} = \int_{P_i} dx\, a_i(x) \int_{P_j} dx'\, C_M(x, x') a_j(x') \tag{10}$$

where $\boldsymbol{x}$ indicates the position along the *i*th ray path (P_i), $\boldsymbol{x}'$ indicates the position along the *j*th ray path (P_j), $a_i(\boldsymbol{x})$ is the data kernel, or weight, for the *i*th path at position $\boldsymbol{x}$, $a_j(\boldsymbol{x})$ is the data kernel for the *j*th path at position $\boldsymbol{x}'$,and $C_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{x}')$ is the model covariance between points $\boldsymbol{x}$ and $\boldsymbol{x}'$ along the two paths *i* and *j*. If $i = j$, then this equation gives variance along a single ray path; if $i \neq j$ then covariance between the ray paths is calculated. Discretizing (10) and transforming from the path specific domain to the SALSA3D grid domain yields

$$\sigma_{\mathrm{ij}} = W_i C_{\mathrm{M}} W_j^T \tag{11}$$

where W_i and W_j are data kernel vectors for the *i*th and *j*th paths, respectively. For a set of arbitrary ray paths, all of the variances and covariances can be calculated at once as

$$\mathrm{C_{TT}} = W C_{\mathrm{M}} W^T \tag{12}$$

where W is the data kernel matrix for all rays to be evaluated with one row per ray path.

Out-Of-Core (OOC) Covariance Matrix Calculation

Traditionally, the use of standard matrix techniques for calculating travel time uncertainty of very large 3D domains has been avoided due to the significant size of the resulting covariance matrix. For example, if the 3D model has 1/2 million distinct grid nodes then the model covariance matrix will contain $5\text{x}10^5$ x $5\text{x}10^5$, or 250 billion elements. If evaluated in double precision the total storage size required will be 2 trillion bytes (~2 TB). While this storage seems significant by standards even a decade ago, modern disk systems can be acquired providing 1 TB of storage for less than $100.00. As a result total storage requirements for problems of the size used in the example above, or smaller, no longer appear to be a limiting factor regarding solution feasibility.

Besides storage issues, the total flop count to evaluate the inverse matrix used to calculate the model covariance matrix can also be enormous. Given that the flop count is proportional to the cube of the node count one would expect more than ~42^{15} Floating Point Operations (FLOP) to perform a Cholesky decomposition of the example matrix above. Given a day to perform the inversion and at least 500 processors working on the problem this total count can be reduced to a per processor rate of 10^9 FLOP per processor per second (1 Giga-FLOP per processor per second). This is well within the current state of the art of modern multi-core processor capability (the example ignores multi-core cache issues which can be significant).

Still, even if we accept that the solution is possible given the current state-of-the-art for disk storage and processing capability we know that the entire problem cannot be held in the modern state-of-the-art computer memory. This is because Terabyte memory systems are still rare, and when available, extremely expensive. However, standard techniques have been developed in the past that allow these types of matrices to be stored in a blocked manner so that each processor only works on a portion of the entire matrix at a time. These techniques are called Out-Of-Core (OOC) solution techniques and have been defined as standard LAPAC routines for many years now (see Gunter 2001).

We can see how OOC methods work by looking at a simple blocked matrix multiplication problem. We know the standard form for the i,j^{th} element of a matrix *c* formed by multiplying two matrices together, say *a* and *b*, is given by

$$c_{i,j} = \sum_{k=0}^{n-1} a_{i,k} b_{k,j}$$

For clarity we'll assume our matrix is symmetric and only 4x4 elements in size. If we block the matrix into blocks of 2x2 elements each we can represent the multiplication as shown in Figure 1 below.

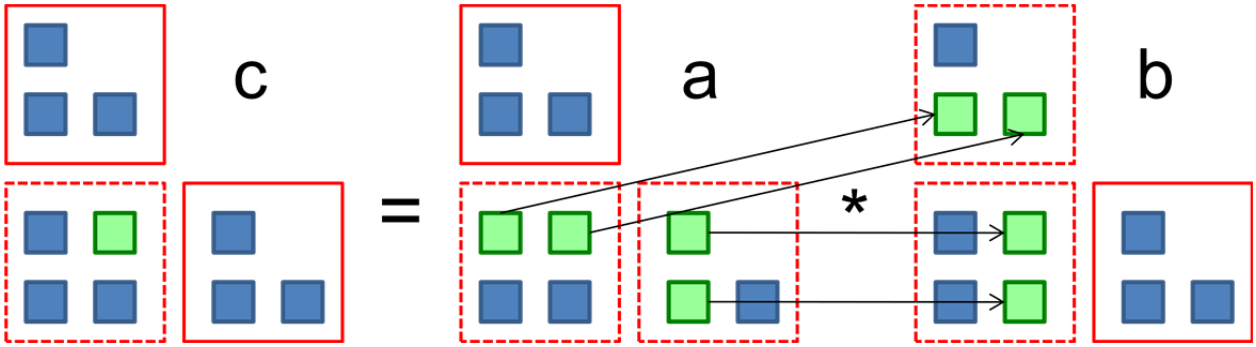

Figure 1. Standard blocked matrix multiply of a 4x4 blocked symmetric matrix using 2x2 blocks. The red dashed block of *c* is updated using the red dashed blocks of *a* and *b*. Each element in a *c* block is composed of an inner product of a row in *a* taken with a row in *b* (e.g. the green elements above).

Note that a single element in **c** can be updated by multiplying all elements in a row of **a** (index *i*) with all elements in a row of **b** (index j). This is a standard definition for matrix multiplication. Further, note that all elements in a single block of **c** can be updated by multiplying similar sub-elements in a row of blocks from matrix **a** with similar sub-elements in a row of blocks from matrix **b**. This means that a block of elements in **c** can be updated by taking the inner product of all elements in a block from matrix **a** with all elements in a block from **b** and summing all contributions from all blocks in a row of **a** with all blocks in a row of **b**.

For this example 2 block multiplies are required to completely update all elements of a single block of **c**. In general, if a matrix is decomposed into an N_b x N_b blocks then determining the final value of any block in **c** requires the sum of N_b such updates.

We have developed an OOC solution methodology that calculates the inverse, $(G^T G)^{-1}$, and subsequently pre- and post-multiplies this result by $G^T \begin{bmatrix} C_{\Delta d} & 0 \\ 0 & C_{\Delta s} \end{bmatrix} G$ to obtain the model covariance matrix C_M. Given the model covariance matrix, the associated uncertainty for any set of arbitrary rays can be calculated from expression (14).

Model Resolution and Traditional Checkerboard Tests

The ability of a tomographic model to resolve features of the model space has historically been assessed using checkerboard tests. In these tests, the velocity field in the model is perturbed from its final tomographic value by some small amount with the sign of the perturbation varying in some regular pattern such as a checkerboard pattern. Then the observed travel times used in tomography are replaced with travel time predictions computed through the perturbed model, and the tomographic inversion is repeated. The velocity model thus obtained can be compared to the actual final tomography model. In areas with good resolution the pattern of the checkerboard perturbations will be accurately retrieved while the pattern will not be discernible in areas with poor resolution. A checkerboard test applied to a recent version of the SALSA3D model at a depth of 820 km is shown in Figure 2a. Figure 2b shows a map of the diagonal elements of the model resolution matrix computed according to equation 7. Both images suggest that the resolution of the model is best in central Europe, the western Pacific and western North America, which correspond to the regions of the model that are best sampled by the available data. The model resolution matrix is more informative however, providing a more quantitative rather than qualitative estimate of the ability of the model to resolve features of the tomographic model.

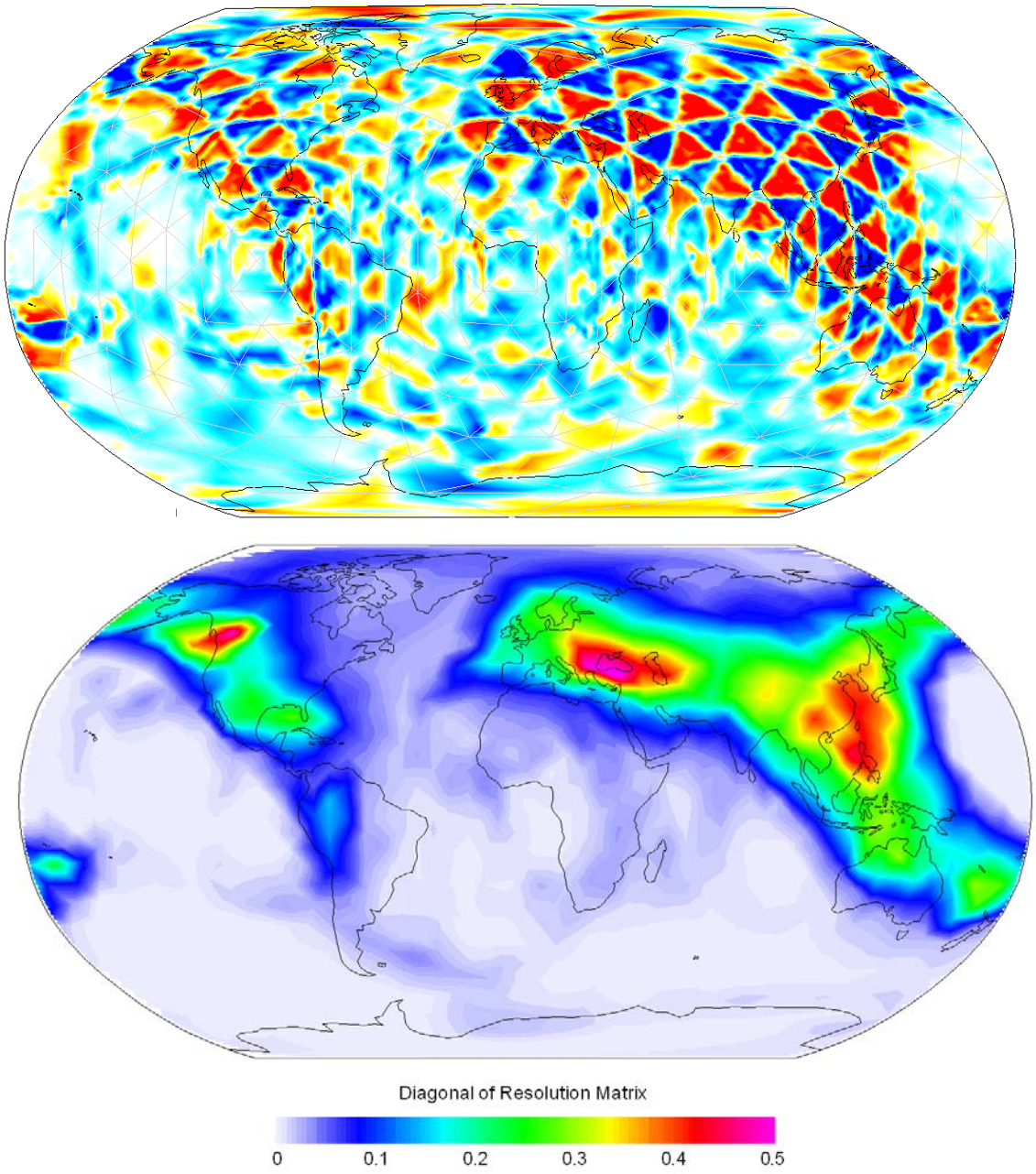

Figure 2 – Evaluating resolution for the SALSA3D model at a depth of 820 km. a) Traditional checkerboard test. b) Diagonal elements of the model resolution matrix.

Variance, Covariance, and Resolution Comparison

Figure 3 shows the diagonal elements of the model covariance matrix at a depth of 820 km. Comparing this to the diagonal elements of the model resolution matrix illustrated in Figure 2 we see that the two are inversely related – in well sampled areas where the model resolution is high, the variance is low and vice versa. The variance values in poorly sampled, high-variance regions are controlled, to a significant degree, by the estimates of delta slowness variance, $\sigma^2_{\Delta s}$, in Equation 9.

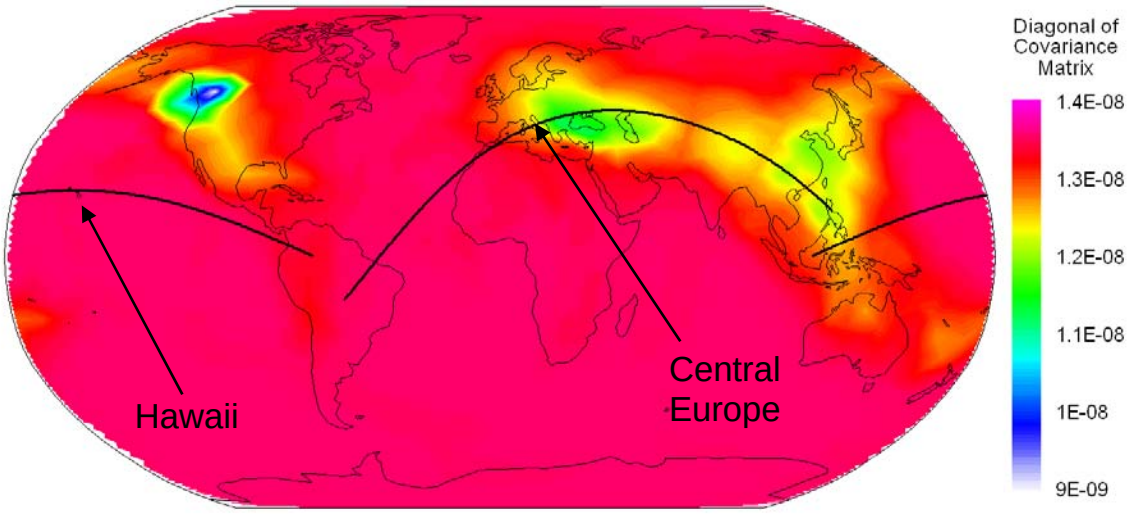

Figure 3 –a) Diagonal of model covariance at 820 km depth. The bold black lines indicate position of cross-section views shown in Figure 4.

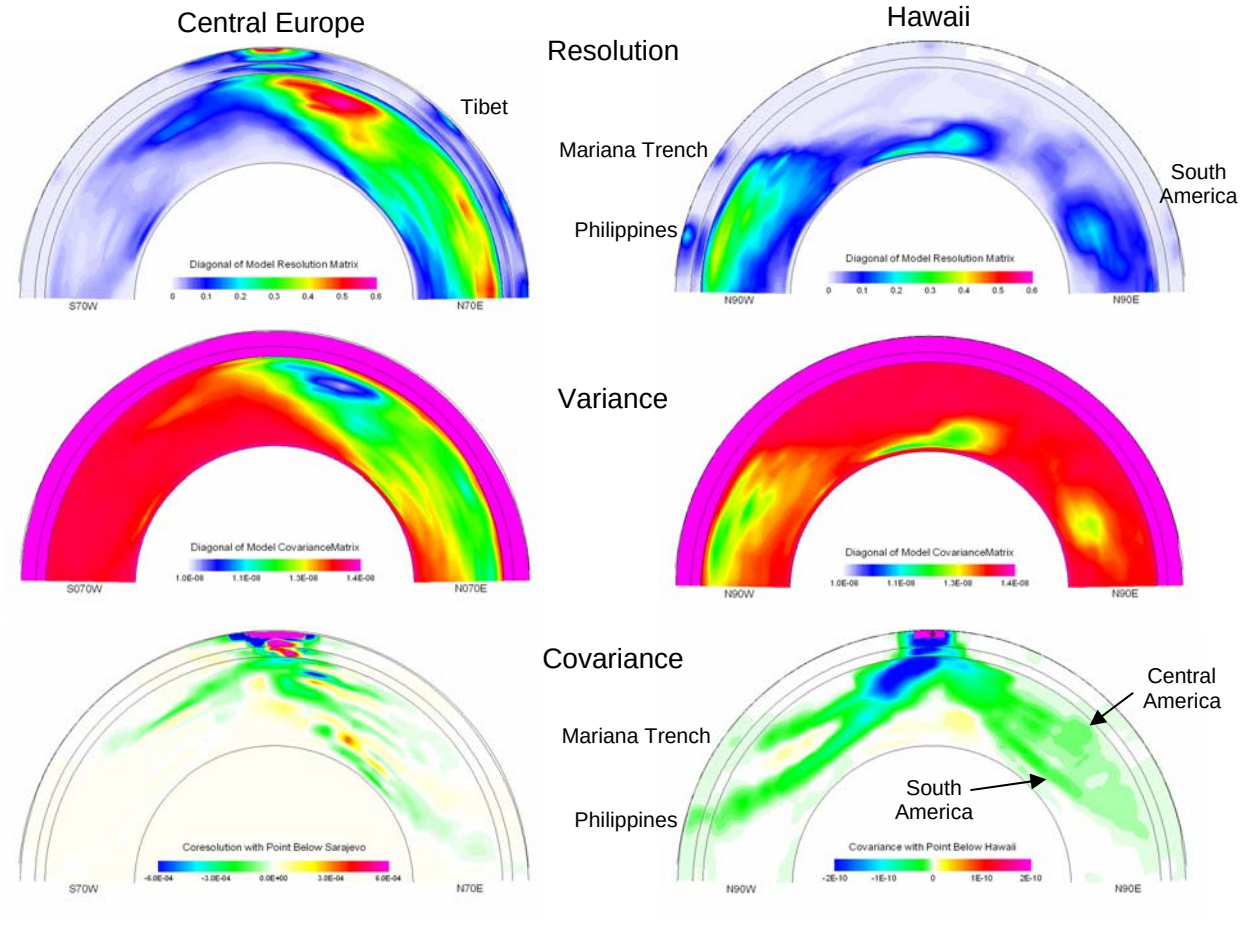

Figure 4 –Resolution, variance, and covariance for the cross-sections shown in Figure 3.

Next we turn our attention to cross-sections through our model. The positions of these cross-sections are shown in Figure 3: one through Sarajevo in central Europe and the other through Hawaii. These present interesting opportunities to assess the implications of the covariance structure of the SALSA3D model. The cross-sections themselves are shown in Figure 4. For each, we show the diagonal of the model resolution matrix, the diagonal of the model covariance matrix (i.e. the variances), and the covariances for nodes located at a depth of 100 km below centers of each map cross-section (i.e. beneath Sarajevo and Hawaii).

Situated in the middle of the Pacific Ocean, Hawaii has stations that observe seismic events around the Pacific Rim but there are few events at intermediate distances from the island. Most importantly, the mantle beneath the Pacific Ocean is poorly sampled by the available data and there are relatively few crossing rays in that part of the model. This is evident in the low values of the diagonal elements of the model resolution matrix beneath the Pacific, including Hawaii. The diagonal elements of the model covariance matrix are the mirror image of the resolution values, with very high variance beneath the Pacific Ocean indicating relatively high uncertainty in the value of P velocity in that portion of the model. For the covariance, we note that the paths through the model that are well sampled, such as the path from Hawaii to the Philippines, are characterized by relatively high negative covariance values compared to paths that are more poorly sampled, suggesting that the model velocity along these paths are not independent. If the velocity at the node below Hawaii were to be increased for some reason, the velocity at other points along the path to the Philippines would have to be reduced in order to compensate. These observations suggest that even though the variance of the mantle velocity along well sampled paths may be very high, the integrated covariance along the path, i.e. the variance of the predicted travel time along the path, may end up being lower that it would be otherwise due to the substantial values of model covariance along the path.

For Sarajevo, we can see that our model is very well-resolved for the portion of the cross-section to the east, hence the model variance is low. To the west, the resolution is poorer, and hence the variance is higher. For the covariance, we can still see negative covariance along ray paths both to the west and the east, suggesting that tradeoffs occur even in relatively well-sampled parts or our model. Integrating covariance along these ray paths will lead to low variance travel times, as would be expected given that this part of the model is well sampled by many crossing ray paths.

Station-Centric Ray Uncertainty

Our last remaining task is to actually calculate travel time uncertainties along specified ray paths using the model covariance matrix, as specified in Equation 12. Again, the memory requirements associated with this calculation mandate an OOC solution, which we have developed. We have succeeded in performing the calculation for a suite of ray paths around a known station location to produce a map of travel time uncertainties, but have found significant numerical artifacts in the solution that we believe are due to poor grid geometries. We are currently experimenting with better distributed model grids to see if these will produce more stable travel time variance results.

CONCLUSIONS AND RECOMMENDATIONS

The use of 3D global tomographic models to predict travel times for locating seismic events seems to be inevitable, yet there are several important problems that must be solved to bring this about. Perhaps foremost among these is the need to be able to produce realistic, path-dependent uncertainties for travel times calculated through these types of models, uncertainty estimates that accurately reflect the very uneven data sampling in the data sets from which these models are derived. In this paper, we have shown how one can first derive the model resolution and model covariance matrices, and from the latter, how path-dependent travel time uncertainties can be estimated. Both the calculation of those matrices and of the travel time uncertainties require huge numbers of calculations and huge amounts of memory making OOC methods a necessity. We have developed such methods and used them with the SNL/LANL SALSA3D model to calculate model resolution and model covariance, and are currently refining our process for calculating travel time uncertainties for specified ray paths.

REFERENCES

Ballard, S., M.A. Begnaud, C.J.Young, J.R. Hipp, M. Chang, A. Enacarnacao, C.A. Rowe, W.S.Phillips, and L.K. Steck (2010). SALSA3D -- a global 3D P-velocity model the Earth's crust and mantle for improved event location, presentation at the Fall 2010 American Geophysical Union Meeting.

Gunter, B.C., and W.C. Reiley (2001). Parallel Out-Of-Core Cholesky and QR Factorizations with POOCLAPACK, 15th International Parallel and Distributed Processing Symposium (IPDPS'01), San Francisco, Ca., USA, September 2001, ISBN: 0-7695-0990-8.

Simmons, N.A., S.C. Myers, and G. Johannesson (2011). The LLNL-3D global P-wave velocity model and its performance in seismic event location, presentation at the 2011 Seismological Society of America Meeting.